

# Multilevel Monte Carlo methods

Michael B. Giles

**Abstract** The author's presentation of multilevel Monte Carlo path simulation at the MCQMC 2006 conference stimulated a lot of research into multilevel Monte Carlo methods. This paper reviews the progress since then, emphasising the simplicity, flexibility and generality of the multilevel Monte Carlo approach. It also offers a few original ideas and suggests areas for future research.

## 1 Introduction

### 1.1 Control variates and two-level MLMC

One of the classic approaches to Monte Carlo variance reduction is through the use of a control variate. Suppose we wish to estimate  $\mathbb{E}[f]$ , and there is a control variate  $g$  which is well correlated to  $f$  and has a known expectation  $\mathbb{E}[g]$ . In that case, we can use the following unbiased estimator for  $\mathbb{E}[f]$ :

$$N^{-1} \sum_{n=1}^N \left\{ f^{(n)} - \lambda \left( g^{(n)} - \mathbb{E}[g] \right) \right\}.$$

The optimal value for  $\lambda$  is  $\rho \sqrt{\mathbb{V}[f] / \mathbb{V}[g]}$ , where  $\rho$  is the correlation between  $f$  and  $g$ , and the variance of the control variate estimator is reduced by factor  $1 - \rho^2$  compared to the standard estimator.

A two-level version of MLMC (multilevel Monte Carlo) is very similar. If we want to estimate  $\mathbb{E}[P_1]$  but it is much cheaper to simulate  $P_0 \approx P_1$ , then since

$$\mathbb{E}[P_1] = \mathbb{E}[P_0] + \mathbb{E}[P_1 - P_0]$$

---

Michael B. Giles

Mathematical Institute, University of Oxford, e-mail: mike.giles@maths.ox.ac.uk

we can use the unbiased two-level estimator

$$N_0^{-1} \sum_{n=1}^{N_0} P_0^{(n)} + N_1^{-1} \sum_{n=1}^{N_1} \left( P_1^{(n)} - P_0^{(n)} \right).$$

Here  $P_1^{(n)} - P_0^{(n)}$  represents the difference between  $P_1$  and  $P_0$  for the same underlying stochastic sample, so that  $P_1^{(n)} - P_0^{(n)}$  is small and has a small variance; the precise construction depends on the application and various examples will be shown later. The two key differences from the control variate approach are that the value of  $\mathbb{E}[P_0]$  is not known, so has to be estimated, and we use  $\lambda = 1$ .

If we define  $C_0$  and  $C_1$  to be the cost of computing a single sample of  $P_0$  and  $P_1 - P_0$ , respectively, then the total cost is  $N_0 C_0 + N_1 C_1$ , and if  $V_0$  and  $V_1$  are the variance of  $P_0$  and  $P_1 - P_0$ , then the overall variance is  $N_0^{-1} V_0 + N_1^{-1} V_1$ , assuming that  $\sum_{n=1}^{N_0} P_0^{(n)}$  and  $\sum_{n=1}^{N_1} \left( P_1^{(n)} - P_0^{(n)} \right)$  use independent samples.

Hence, treating the integers  $N_0, N_1$  as real variables and performing a constrained minimisation using a Lagrange multiplier, the variance is minimised for a fixed cost by choosing  $N_1/N_0 = \sqrt{V_1/C_1} / \sqrt{V_0/C_0}$ .

## 1.2 Multilevel Monte Carlo

The full multilevel generalisation is quite natural: given a sequence  $P_0, P_1, \dots$ , which approximates  $P_L$  with increasing accuracy, but also increasing cost, we have the simple identity

$$\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{\ell=1}^L \mathbb{E}[P_\ell - P_{\ell-1}],$$

and therefore we can use the following unbiased estimator for  $\mathbb{E}[P_L]$ ,

$$N_0^{-1} \sum_{n=1}^{N_0} P_0^{(0,n)} + \sum_{\ell=1}^L \left\{ N_\ell^{-1} \sum_{n=1}^{N_\ell} \left( P_\ell^{(\ell,n)} - P_{\ell-1}^{(\ell,n)} \right) \right\}$$

with the inclusion of the level  $\ell$  in the superscript  $(\ell, n)$  indicating that the samples used at each level of correction are independent.

If we define  $C_0, V_0$  to be the cost and variance of one sample of  $P_0$ , and  $C_\ell, V_\ell$  to be the cost and variance of one sample of  $P_\ell - P_{\ell-1}$ , then the overall cost and variance of the multilevel estimator is  $\sum_{\ell=0}^L N_\ell C_\ell$  and  $\sum_{\ell=0}^L N_\ell^{-1} V_\ell$ , respectively.

For a fixed cost, the variance is minimised by choosing  $N_\ell = \lambda \sqrt{V_\ell / C_\ell}$  for some value of the Lagrange multiplier  $\lambda$ . In particular, to achieve an overall variance of  $\varepsilon^2$  requires that  $\lambda = \varepsilon^{-2} \sum_{\ell=0}^L \sqrt{V_\ell C_\ell}$ . The total computational cost is then

$$C = \varepsilon^{-2} \left( \sum_{\ell=0}^L \sqrt{V_\ell C_\ell} \right)^2. \quad (1)$$

It is important to note whether the product  $V_\ell C_\ell$  increases or decreases with  $\ell$ , i.e. whether or not the cost increases with level faster than the variance decreases. If it increases with level, so that the dominant contribution to the cost comes from  $V_L C_L$  then we have  $C \approx \varepsilon^{-2} V_L C_L$ , whereas if it decreases and the dominant contribution comes from  $V_0 C_0$  then  $C \approx \varepsilon^{-2} V_0 C_0$ . This contrasts to the standard MC cost of approximately  $\varepsilon^{-2} V_0 C_L$ , assuming that the cost of computing  $P_L$  is similar to the cost of computing  $P_L - P_{L-1}$ , and that  $\mathbb{V}[P_L] \approx \mathbb{V}[P_0]$ . This shows that in the first case the MLMC cost is reduced by factor  $V_L/V_0$ , corresponding to the ratio of the variances  $\mathbb{V}[P_L - P_{L-1}]$  and  $\mathbb{V}[P_L]$ , whereas in the second case it is reduced by factor  $C_0/C_L$ , the ratio of the costs of computing  $P_0$  and  $P_L - P_{L-1}$ . If the product  $V_\ell C_\ell$  does not vary with level, then the total cost is  $\varepsilon^{-2} L^2 V_0 C_0 = \varepsilon^{-2} L^2 V_L C_L$ .

### 1.3 Earlier related work

Prior to the author's first publications [20, 21] on MLMC for Brownian path simulations, Heinrich developed a multilevel Monte Carlo method for parametric integration, the evaluation of functionals arising from the solution of integral equations, and weakly singular integral operators [33, 34, 35, 36, 37]. Parametric integration concerns the estimation of  $\mathbb{E}[f(x, \lambda)]$  where  $x$  is a finite-dimensional random variable and  $\lambda$  is a parameter. In the simplest case in which  $\lambda$  is a real variable in the range  $[0, 1]$ , having estimated the value of  $\mathbb{E}[f(x, 0)]$  and  $\mathbb{E}[f(x, 1)]$ , one can use  $\frac{1}{2}(f(x, 0) + f(x, 1))$  as a control variate when estimating the value of  $\mathbb{E}[f(x, \frac{1}{2})]$ . This approach can then be applied recursively for other intermediate values of  $\lambda$ , yielding large savings if  $f(x, \lambda)$  is sufficiently smooth with respect to  $\lambda$ . Although this does not quite fit into the general MLMC form given in the previous section, the recursive control variate approach is very similar and the complexity analysis is also very similar to the analysis to be presented in the next section.

Although not so clearly related, there are papers by Brandt *et al* [9, 10] which combine Monte Carlo techniques with multigrid ideas in determining thermodynamic limits in statistical physics applications. It is the multigrid ideas of Brandt and others for the iterative solution of systems of equations which were the inspiration for the author in developing the MLMC method for SDE path simulation.

In 2005, Kebaier [41] developed a two-level approach for path simulation which is very similar to the author's approach presented in the next section. The only differences are the use of only two levels, and the use of a general multiplicative factor as in the standard control variate approach. A similar multilevel approach was under development at the same time by Speight, but was not published until later [49, 50].

## 2 MLMC theorem

In the Introduction, we considered the case of a general multilevel method in which the output  $P_L$  on the finest level corresponds to the quantity of interest. However, in many infinite-dimensional applications, such as in SDEs and SPDEs, the output  $P_\ell$  on level  $\ell$  is an approximation to a random variable  $P$ . In this case, the mean square error (MSE) has the usual decomposition into the total variance of the multilevel estimator, plus the square of the bias  $(\mathbb{E}[P_L - P])^2$ . To achieve an MSE which is less than  $\varepsilon^2$ , it is sufficient to ensure that each of these terms is less than  $\frac{1}{2}\varepsilon^2$ . This leads to the following theorem:

**Theorem 1.** *Let  $P$  denote a random variable, and let  $P_\ell$  denote the corresponding level  $\ell$  numerical approximation.*

*If there exist independent estimators  $Y_\ell$  based on  $N_\ell$  Monte Carlo samples, and positive constants  $\alpha, \beta, \gamma, c_1, c_2, c_3$  such that  $\alpha \geq \frac{1}{2} \min(\beta, \gamma)$  and*

- i)  $|\mathbb{E}[P_\ell - P]| \leq c_1 2^{-\alpha\ell}$
- ii)  $\mathbb{E}[Y_\ell] = \begin{cases} \mathbb{E}[P_0], & \ell = 0 \\ \mathbb{E}[P_\ell - P_{\ell-1}], & \ell > 0 \end{cases}$
- iii)  $\mathbb{V}[Y_\ell] \leq c_2 N_\ell^{-1} 2^{-\beta\ell}$
- iv)  $\mathbb{E}[C_\ell] \leq c_3 N_\ell 2^{\gamma\ell}$ , where  $C_\ell$  is the computational complexity of  $Y_\ell$

*then there exists a positive constant  $c_4$  such that for any  $\varepsilon < e^{-1}$  there are values  $L$  and  $N_\ell$  for which the multilevel estimator*

$$Y = \sum_{\ell=0}^L Y_\ell,$$

*has a mean-square-error with bound*

$$MSE \equiv \mathbb{E} \left[ (Y - \mathbb{E}[P])^2 \right] < \varepsilon^2$$

*with a computational complexity  $C$  with bound*

$$\mathbb{E}[C] \leq \begin{cases} c_4 \varepsilon^{-2}, & \beta > \gamma, \\ c_4 \varepsilon^{-2} (\log \varepsilon)^2, & \beta = \gamma, \\ c_4 \varepsilon^{-2 - (\gamma - \beta)/\alpha}, & \beta < \gamma. \end{cases}$$

The statement of the theorem is a slight generalisation of the original theorem in [21]. It corresponds to the theorem and proof in [15], except for the minor change to expected costs to allow for applications such as jump-diffusion modelling in which the simulation cost of individual samples is itself random.

The theorem is based on the idea of a geometric progression in the levels of approximation, leading to the exponential decay in the weak error in condition i), and

the variance in condition *iii*), as well as the exponential increase in the expected cost in condition *iv*). This geometric progression was based on experience with multigrid methods in the iterative solution of large systems of linear equations, but it is worth noting that it is not necessarily the optimal choice in all circumstances.

The result of the theorem merits some discussion. In the case  $\beta > \gamma$ , the dominant computational cost is on the coarsest levels where  $C_\ell = O(1)$  and  $O(\varepsilon^{-2})$  samples are required to achieve the desired accuracy. This is the standard result for a Monte Carlo approach using i.i.d. samples; to do better would require an alternative approach such as the use of Latin hypercube sampling or quasi-Monte Carlo methods. In the case  $\beta < \gamma$ , the dominant computational cost is on the finest levels. Because of condition *i*),  $2^{-\alpha L} = O(\varepsilon)$ , and hence  $C_L = O(\varepsilon^{-\gamma/\alpha})$ . If  $\beta = 2\alpha$ , which is usually the largest possible value for a given  $\alpha$ , for reasons explained below, then the total cost is  $O(C_L)$  corresponding to  $O(1)$  samples on the finest level, again the best that can be achieved. The dividing case  $\beta = \gamma$  is the one for which both the computational effort, and the contributions to the overall variance, are spread approximately evenly across all of the levels; the  $(\log \varepsilon)^2$  term corresponds to the  $L^2$  factor in the corresponding discussion in section 1.2.

The natural choice for the multilevel estimator is

$$Y_\ell = N_\ell^{-1} \sum_i P_\ell(\omega_i) - P_{\ell-1}(\omega_i), \quad (2)$$

where  $P_\ell(\omega_i)$  is the approximation to  $P(\omega_i)$  on level  $\ell$ , and  $P_{\ell-1}(\omega_i)$  is the corresponding approximation on level  $\ell-1$  for the same underlying stochastic sample  $\omega_i$ . Note that  $\mathbb{V}[P_\ell - P_{\ell-1}]$  is usually similar in magnitude to  $\mathbb{E}[(P_\ell - P_{\ell-1})^2]$  which is greater than  $(\mathbb{E}[P_\ell - P_{\ell-1}])^2$ ; this implies that  $\beta \leq 2\alpha$  and hence the condition in the theorem that  $\alpha \geq \frac{1}{2} \min(\beta, \gamma)$  is satisfied.

However, the multilevel theorem allows for the use of other estimators, provided they satisfy the restriction of condition *ii*) which ensures that  $\mathbb{E}[Y] = \mathbb{E}[P_L]$ . Two examples of this will be given later in the paper. In the first, slightly different numerical approximations are used for the coarse and fine paths in SDE simulations, giving

$$Y_\ell = N_\ell^{-1} \sum_i P_\ell^f(\omega_i) - P_{\ell-1}^c(\omega_i).$$

Provided  $\mathbb{E}[P_\ell^f] = \mathbb{E}[P_\ell^c]$  so that the expectation on level  $\ell$  is the same for the two approximations, then condition *ii*) is satisfied and no additional bias (other than the bias due to the approximation on the finest level) is introduced into the multilevel estimator. The second example defines an antithetic  $\omega_i^a$  with the same distribution as  $\omega_i$ , and then uses the multilevel estimator

$$Y_\ell = N_\ell^{-1} \sum_i \frac{1}{2} (P_\ell(\omega_i) + P_\ell(\omega_i^a)) - P_{\ell-1}(\omega_i).$$

Since  $\mathbb{E}[P_\ell(\omega_i^a)] = \mathbb{E}[P_\ell(\omega_i)]$ , then again condition *ii*) is satisfied. In each case, the objective in constructing a more complex estimator is to achieve a greatly reduced variance  $\mathbb{V}[Y_\ell]$  so that fewer samples are required.

### 3 SDEs

#### 3.1 Euler discretisation

The original multilevel path simulation paper [21] treated SDEs using the simple Euler-Maruyama discretisation together with the natural multilevel estimator (2).

Provided the SDE satisfies the usual conditions (see Theorem 10.2.2 in [42]), the strong error for the Euler discretisation with timestep  $h$  is  $O(h^{1/2})$ , and therefore for Lipschitz payoff functions  $P$  (such as European, Asian and lookback options in finance) the variance  $V_\ell \equiv \mathbb{V}[P_\ell - P_{\ell-1}]$  is  $O(h_\ell)$ . If  $h_\ell = 4^{-\ell}h_0$ , as in [21], then this gives  $\alpha=2$ ,  $\beta=4$  and  $\gamma=2$ . Alternatively, if  $h_\ell = 2^{-\ell}h_0$ , then  $\alpha=1$ ,  $\beta=2$  and  $\gamma=1$ . In either case, Theorem 1 gives the complexity to achieve a root-mean-square error of  $\varepsilon$  to be  $O(\varepsilon^{-2}(\log \varepsilon)^2)$ , which is near-optimal as Müller-Gronbach & Ritter have proved an  $O(\varepsilon^{-2})$  lower bound for the complexity [46].

For other payoff functions the complexity is higher.  $V_\ell \approx O(h^{1/2})$  for the digital option which is a discontinuous function of the SDE solution at the final time, and the barrier option which depends discontinuously on the minimum or maximum value over the full time interval. Loosely speaking, this is because there is an  $O(h^{1/2})$  probability of the coarse and fine paths being on opposite sides of the discontinuity, and in such cases there is an  $O(1)$  difference in the payoff. Currently, there is no known “fix” for this for the Euler-Maruyama discretisation; we will return to this issue for the Milstein discretisation when there are ways of improving the situation.

Table 1 summarises the observed variance convergence rate in numerical experiments for the different options, and the theoretical results which have been obtained; the digital option analysis is due to Avikainen [4] while the others are due to Giles, Higham & Mao [24]. Although the analysis in some of these cases is for one-dimensional SDEs, it also applies to multi-dimensional SDEs [22].

option	Euler-Maruyama		Milstein	
	numerics	analysis	numerics	analysis
Lipschitz	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
Asian	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
lookback	$O(h)$	$O(h)$	$O(h^2)$	$o(h^{2-\delta})$
barrier	$O(h^{1/2})$	$o(h^{1/2-\delta})$	$O(h^{3/2})$	$o(h^{3/2-\delta})$
digital	$O(h^{1/2})$	$O(h^{1/2} \log h)$	$O(h^{3/2})$	$o(h^{3/2-\delta})$

**Table 1** Observed and theoretical convergence rates for the multilevel correction variance for scalar SDEs, using the Euler-Maruyama and Milstein discretisations.  $\delta$  is any strictly positive constant.

### 3.2 Milstein discretisation

For Lipschitz payoffs, the variance  $V_\ell$  for the natural multilevel estimator converges at twice the order of the strong convergence of the numerical approximation of the SDE. This immediately suggests that it would be better to replace the Euler-Maruyama discretisation by the Milstein discretisation [20] since it gives first order strong convergence under certain conditions (see Theorem 10.3.5 in [42]).

This immediately gives an improved variance for European and Asian options, as shown in Table 1, but to get the improved variance for lookback, barrier and digital options requires the construction of estimators which are slightly different on the coarse and fine path simulations, but which respect the condition that  $\mathbb{E}[P_\ell^f] = \mathbb{E}[P_\ell^c]$ .

The construction for the digital option will be discussed next, but for the lookback and barrier options, the key is the definition of a Brownian Bridge interpolant based on the approximation that the drift and volatility do not vary within the timestep. For each coarse timestep, the mid-point of the interpolant can be sampled using knowledge of the fine path Brownian increments, and then classical results can be used for the distribution of the minimum or maximum within each fine timestep for both the fine and coarse path approximations [29]. The full details are given in [20], and Table 1 summarises the convergence behaviour observed numerically, and the supporting numerical analysis by Giles, Debrabant & Röbler [23].

The outcome is that for the case in which the number of timesteps doubles at each level, so  $h_\ell = 2^{-\ell}h_0$ , then  $\gamma = 1$  and either  $\beta = 2$  (European, Asian and lookback) or  $\beta = 1.5$  (barrier and digital). Hence, we are in the regime where  $\beta > \gamma$  and the overall complexity is  $O(\varepsilon^{-2})$ . Furthermore, the dominant computational cost is on the coarsest levels of simulation.

Since the coarsest levels are low-dimensional, they are well suited to the use of quasi-Monte Carlo methods which are particularly effective in lower dimensions because of the existence of  $O((\log N)^d/N)$  error bounds, where  $d$  is the dimension and  $N$  is the number of QMC points. The bounds are for the numerical integration of certain function classes on the unit hypercube, and are a consequence of the Koksma-Hlawka inequality together with bounds on the star-discrepancy of certain sequences of QMC points.

This has been investigated by Giles & Waterhouse [28] using a rank-1 lattice rule to generate the quasi-random numbers, randomisation with 32 independent offsets to obtain confidence intervals, and a standard Brownian Bridge construction of the increments of the driving Brownian process. The numerical results show that MLMC on its own was better than QMC on its own, but the combination of the two was even better. The QMC treatment greatly reduced the variance per sample for the coarsest levels, resulting in significantly reduced costs overall. In the simplest case of a Lipschitz European payoff, the computational complexity was reduced from  $O(\varepsilon^{-2})$  to approximately  $O(\varepsilon^{-1.5})$ .

### 3.2.1 Digital options

As discussed earlier, discontinuous payoffs pose a challenge to the multilevel Monte Carlo approach, because small differences in the coarse and fine path simulations can lead to an  $O(1)$  difference in the payoff function. This leads to a slower decay in the variance  $V_\ell$ , and because the fourth moment is also much larger it leads to more samples being required to obtain an accurate estimate for  $V_\ell$ , which is needed to determine the optimal number of samples  $N_\ell$ .

This is a generic problem. Although we will discuss it here in the specific context of a Brownian SDE and an option which is a discontinuous function of the underlying at the final time, the methods which are discussed are equally applicable in a range of other cases. Indeed, some of these techniques have been first explored in the context of pathwise sensitivity analysis [12] or jump-diffusion modelling [52].

#### Conditional expectation

The conditional expectation approach builds on a well-established technique for payoff smoothing which is used for pathwise sensitivity analysis (see, for example, pp. 399-400 in [29]).

We start by considering the fine path simulation, and make a slight change by using the Euler-Maruyama discretisation for the final timestep, instead of the Milstein discretisation. Conditional on the numerical approximation of the value  $S_{T-h}$  one timestep before the end (which in turn depends on all of the Brownian increments up to that time) the numerical approximation for the final value  $S_T$  now has a Gaussian distribution, and for a simple digital option the conditional expectation is known analytically.

The same treatment is used for the coarse path, except that in the final timestep, we re-use the known value of the Brownian increment for the second last fine timestep, which corresponds to the first half of the final coarse timestep. This results in the conditional distribution for the coarse path underlying at maturity matching that of the fine path to within  $O(h)$ , for both the mean and the standard deviation [23]. Consequently, the difference in payoff between the coarse and fine paths near the payoff discontinuity is  $O(h^{1/2})$ , and so the variance is approximately  $O(h^{3/2})$ .

#### Splitting

The conditional expectation technique works well in 1D where there is a known analytic value for the conditional expectation, but in multiple dimensions it may not be known. In this case, one can use the technique of “splitting” [3]. Here the conditional expectation is replaced by a numerical estimate, averaging over a number of sub-samples. i.e. for each set of Brownian increments up to one fine timestep before the end, one uses a number of samples of the final Brownian increment to produce an average payoff. If the number of sub-samples is chosen appropriately, the variance is the same, to leading order, without any increase in the computational cost, again to leading order. Because of its simplicity and generality, this is now my preferred approach. Furthermore, one can revert to using the Milstein approximation for the final timestep.



### Change of measure

The change of measure approach is another approximation to the conditional expectation. The fine and coarse path conditional distributions at maturity are two very similar Gaussian distributions. Instead of following the splitting approach of taking corresponding samples from these two distributions, we can instead take a sample from a third Gaussian distribution (with a mean and variance perhaps equal to the average of the other two). This leads to the introduction of a Radon-Nikodym derivative for each path, and the difference in the payoffs from the two paths is then due to the difference in their Radon-Nikodym derivatives.

In the specific context of digital options, this is a more complicated method to implement, and the resulting variance is no better. However, in other contexts a similar approach can be very effective.

### 3.2.2 Multi-dimensional SDEs

The discussion so far has been for scalar SDEs, but the computational benefits of Monte Carlo methods arise in higher dimensions. For multi-dimensional SDEs satisfying the usual commutativity condition (see, for example, p.353 in [29]) the Milstein discretisation requires only Brownian increments for its implementation, and most of the analysis above carries over very naturally.

The only difficulties are in lookback and barrier options where the classical results for the distribution of the minimum or maximum of a one-dimensional Brownian motion, do not extend to the joint distribution of the minima or maxima of two correlated Brownian motions. An alternative approach may be to sub-sample from the Brownian Bridge interpolant for those timesteps which are most likely to give the global minimum or maximum. This may need to be combined with splitting for the barrier option to avoid the  $O(1)$  difference in payoffs. An alternative might be to use adaptive time-stepping [40].

For multi-dimensional SDEs which do not satisfy the commutativity condition the Milstein discretisation requires the simulation of Lévy areas. This is unavoidable to achieve first order strong convergence; the classical result of Clark & Cameron says that  $O(h^{1/2})$  strong convergence is the best that can be achieved in general using just Brownian increments [14].

However, Giles & Lukasz have developed an antithetic treatment which achieves a very low variance despite the  $O(h^{1/2})$  strong convergence [26]. The estimator which is used is

$$Y_\ell = N_\ell^{-1} \sum_i \frac{1}{2} (P_\ell(\omega_i) + P_\ell(\omega_i^a)) - P_{\ell-1}(\omega_i).$$

Here  $\omega_i$  represents the driving Brownian path, and  $\omega_i^a$  is an antithetic counterpart defined by a time-reversal of the Brownian path within each coarse timestep. This results in the Brownian increments for the antithetic fine path being swapped relative to the original path. Lengthy analysis proves that the average of the fine and

antithetic paths is within  $O(h)$  of the coarse path, and hence the multilevel variance is  $O(h^2)$  for smooth payoffs, and  $O(h^{3/2})$  for the standard European call option.

This treatment has been extended to handle lookback and barrier options [27]. This combines sub-sampling of the Brownian path to approximate the Lévy areas with sufficient accuracy to achieve  $O(h^{3/4})$  strong convergence, with an antithetic treatment at the finest level of resolution to ensure that the average of the fine paths is within  $O(h)$  of the coarse path.

### 3.3 Lévy processes

#### 3.3.1 Jump-diffusion processes

With finite activity jump-diffusion processes, such as in the Merton model [44], it is natural to simulate each individual jump using a jump-adapted discretisation [47].

If the jump rate is constant, then the jumps on the coarse and fine paths will occur at the same time, and the extension of the multilevel method is straightforward [52].

If the jump rate is path-dependent then the situation is trickier. If there is a known upper bound to the jump rate, then one can use Glasserman & Merener's "thinning" approach [31] in which a set of candidate jump times is simulated based on the constant upper bound, and then a subset of these are selected to be real jumps. The problem with the multilevel extension of this is that some candidate jumps will be selected for the coarse path but not for the fine path, or vice versa, leading to an  $O(1)$  difference in the paths and hence the payoffs. Xia overcomes this by using a change of measure to select the jump times consistently for both paths, with a Radon-Nikodym derivative being introduced in the process [52].

#### 3.3.2 More general processes

With infinite activity Lévy processes it is impossible to simulate each jump. One approach is to simulate the large jumps and either neglect the small jumps or approximate their effect by adding a Brownian diffusion term [17, 18, 43]. Following this approach, the cutoff  $\delta_\ell$  for the jumps which are simulated varies with level, and  $\delta_\ell \rightarrow 0$  as  $\ell \rightarrow \infty$  to ensure that the bias converges to zero. In the multilevel treatment, when simulating  $P_\ell - P_{\ell-1}$  the jumps fall into three categories. The ones which are larger than  $\delta_{\ell-1}$  get simulated in both the fine and coarse paths. The ones which are smaller than  $\delta_\ell$  are either neglected for both paths, or approximated by the same Brownian increment. The difficulty is in the intermediate range  $[\delta_\ell, \delta_{\ell-1}]$  in which the jumps are simulated for the fine path, but neglected or approximated for the coarse path. This is what leads to the difference in path simulations, and hence to a non-zero value for  $P_\ell - P_{\ell-1}$ .

Alternatively, for many SDEs driven by a Lévy process it is possible to directly simulate the increments of the Lévy process over a set of uniform timesteps [16, 48],

in exactly the same way as one simulates Brownian increments. For other Lévy processes, it may be possible in the future to simulate the increments by constructing approximations to the inverse of the cumulative distribution function. Where this is possible, it may be the best approach to achieve a close coupling between the coarse and fine path simulations, and hence a low variance  $V_\ell$ , since the increments of the driving Lévy process for the coarse path can be obtained trivially by summing the increments for the fine path.

## 4 SPDEs

After developing the MLMC method for SDE simulations, it was immediately clear that it was equally applicable to SPDEs, and indeed the computational savings would be greater because the cost of a single sample increases more rapidly with grid resolution for SPDEs with higher space-time dimension.

In 2006, the author discussed this with Thomas Hou in the specific context of elliptic SPDEs with random coefficients, and Hou's postdoc then performed the first unpublished MLMC computations for SPDEs. The first published work was by a student of Klaus Ritter in her Diploma thesis [32]; her application was to parabolic SPDEs. Since this early work, there has been a variety of papers on elliptic [6, 13, 15, 51], parabolic [5, 25] and hyperbolic [45] SPDEs.

In almost all of this work, the construction of the multilevel estimator is quite natural, using a geometric sequence of grids and the usual estimators for  $P_\ell - P_{\ell-1}$ . It is the numerical analysis of the variance of the multilevel estimator which is often very challenging.

### 4.1 Elliptic SPDE

The largest amount of research on multilevel for SPDEs has been for elliptic PDEs with random coefficients. The PDE typically has the form

$$-\nabla \cdot (k(\mathbf{x}, \omega) \nabla p(\mathbf{x}, \omega)) = 0, \quad \mathbf{x} \in D.$$

with Dirichlet or Neumann boundary conditions on the boundary  $\partial D$ . For subsurface flow problems, such as the modelling of groundwater flow in nuclear waste repositories, the diffusivity (or permeability)  $k$  is often modelled as a lognormal random field, i.e.  $\log k$  is a Gaussian field with a uniform mean (which we will take to be zero for simplicity) and a covariance function of the general form  $R(\mathbf{x}, \mathbf{y}) = r(\mathbf{x} - \mathbf{y})$ . Samples of  $\log k$  are provided by a Karhunen-Loève expansion:

$$\log k(\mathbf{x}, \omega) = \sum_{n=0}^{\infty} \sqrt{\theta_n} \xi_n(\omega) f_n(\mathbf{x}),$$

where  $\theta_n$  are the eigenvalues of  $R(\mathbf{x}, \mathbf{y})$  in decreasing order,  $f_n$  are the corresponding eigenfunctions, and  $\xi_n$  are independent unit Normal random variables. However, it is more efficient to generate them using a circulant embedding technique which enables the use of FFTs [19].

The multilevel treatment is straightforward. The spatial grid resolution is doubled on each level. Using the Karhunen-Loève generation, the expansion is truncated after  $K_\ell$  terms, with  $K_\ell$  increasing with level [51]; in unpublished work, a similar approach has also been used with the circulant embedding generation.

In both cases,  $\log k$  is generated using a row-vector of independent unit Normal random variables  $\xi$ . The variables for the fine level can be partitioned into those for the coarse level  $\xi_{\ell-1}$ , plus some additional variables  $z_\ell$ , giving  $\xi_\ell = (\xi_{\ell-1}, z_\ell)$ . It is possible to develop an antithetic treatment similar to that used for SDEs by defining  $\xi_\ell^a = (\xi_{\ell-1}, -z_\ell)$ . This gives a second  $\log k_\ell^a$  field on the fine grid, and then the multilevel estimator can be based on the average of the two outputs obtained on the fine grid, minus the output obtained on the coarse grid using  $\log k_{\ell-1}$ . Unfortunately, numerical experiments indicate it gives little benefit; it is mentioned here as another illustration of an antithetic estimator, and as a warning that it does not always yields significant benefits.

The numerical analysis of the multilevel approach for these elliptic SPDE applications is challenging because the diffusivity is unbounded, but Charrier, Scheichl & Teckentrup [13] have successfully analysed it for certain output functionals, and Teckentrup *et al* have further developed the analysis for other output functionals and more general log-normal diffusivity fields [51].

## 4.2 Parabolic SPDE

Giles & Reisinger [25] consider an unusual SPDE from credit default modelling,

$$dp = -\mu \frac{\partial p}{\partial x} dt + \frac{1}{2} \frac{\partial^2 p}{\partial x^2} dt - \sqrt{\rho} \frac{\partial p}{\partial x} dM_t, \quad x > 0$$

subject to boundary condition  $p(0, t) = 0$ . Here  $p(x, t)$  represents the probability density function for firms being a distance  $x$  from default at time  $t$ . The diffusive term is due to idiosyncratic factors affecting individual firms, while the stochastic term due to the scalar Brownian motion  $M_t$  corresponds to the systemic movement due to random market effects affecting all firms. The payoff corresponds to different tranches of a credit derivative which depends on the integral  $\int_0^\infty p(x, t) dx$  at a set of discrete times.

A Milstein time discretisation with timestep  $k$ , and a central space discretisation of the spatial derivatives with uniform spacing  $h$  gives the numerical approximation

$$p_j^{n+1} = p_j^n - \frac{\mu k + \sqrt{\rho} k Z_n}{2h} (p_{j+1}^n - p_{j-1}^n) + \frac{(1-\rho)k + \rho k Z_n^2}{2h^2} (p_{j+1}^n - 2p_j^n + p_{j-1}^n)$$

where  $p_j^n \approx p(jh, nk)$ , and the  $Z_n$  are standard Normal random variables so that  $\sqrt{h} Z_n$  corresponds to an increment of the driving scalar Brownian motion.

The multilevel implementation is very straightforward, with  $k_\ell = k_{\ell-1}/2$  and  $h_\ell = h_{\ell-1}/4$  due to numerical stability considerations which are analysed in the paper. As with SDEs, the coupling between the coarse and fine samples comes from summing the fine path Brownian increments in pairs to give the increments for the coarse path. The computational cost increases by factor 8 on each level, and numerical experiments indicate that the variance decreases by factor 8, so the overall computational complexity to achieve an  $O(\varepsilon)$  RMS error is again  $O(\varepsilon^{-2}(\log \varepsilon)^2)$ .

## 5 Continuous-time Markov Chain simulation

Anderson & Higham have recently developed a very interesting new application of multilevel to continuous-time Markov Chain simulation [2]. Although they present their work in the context of stochastic chemical reactions, when species concentrations are extremely low and so stochastic effects become significant, they point out that the method has wide applicability in other areas.

In the simplest case of a single chemical reaction, the ‘‘tau-leaping’’ method (which is essentially the Euler-Maruyama method, approximating the reaction rate as being constant throughout the timestep) gives the discrete equation

$$\mathbf{x}_{n+1} = \mathbf{x}_n + P(h \lambda(\mathbf{x}_n)),$$

where  $h$  is the timestep,  $\lambda(\mathbf{x}_n)$  is the reaction rate (or propensity function), and  $P(t)$  represents a unit-rate Poisson random variable over time interval  $t$ .

If this equation defines the fine path in the multilevel simulation, then the coarse path, with double the timestep, is given by

$$\mathbf{x}_{n+2}^c = \mathbf{x}_n^c + P(2h \lambda(\mathbf{x}_n^c))$$

for even timesteps  $n$ . The question then is how to couple the coarse and fine path simulations.

The key observation by Anderson & Higham [2] is that for any  $t_1, t_2 > 0$ , the sum of two independent Poisson variates  $P(t_1), P(t_2)$  is equivalent in distribution to  $P(t_1+t_2)$ . Based on this, the first step is to express the coarse path Poisson variate as the sum of two Poisson variates,  $P(h \lambda(\mathbf{x}_n^c))$  corresponding to the first and second fine path timesteps. For the first of the two fine timesteps, the coarse and fine path Poisson variates are coupled by defining two Poisson variates based on the minimum of the two reactions rates, and the absolute difference,

$$P_1 = P\left(h \min(\lambda(\mathbf{x}_n), \lambda(\mathbf{x}_n^c))\right), \quad P_2 = P\left(h |\lambda(\mathbf{x}_n) - \lambda(\mathbf{x}_n^c)|\right),$$

and then using  $P_1$  as the Poisson variate for the path with the smaller rate, and  $P_1 + P_2$  for the path with the larger rate. This elegant approach naturally gives a small difference in the Poisson variates when the difference in rates is small, and leads to a very effective multilevel algorithm.

In their paper [2], Anderson & Higham treat more general systems with multiple reactions, and include an additional coupling at the finest level to an SSA (Stochastic Simulation Algorithm) computation, so that their overall multilevel estimator is unbiased, unlike the estimators discussed earlier for SDEs. Finally, they give a complete numerical analysis of the variance of their multilevel algorithm.

Because stochastic chemical simulations typically involve 1000's of reactions, the multilevel method is particularly effective in this context, providing computational savings in excess of a factor of 100 [2].

## 6 Wasserstein metric

In the multilevel treatment of SDEs, the Brownian or Lévy increments for the coarse path are obtained by summing the increments for the fine path. Similarly, in the Markov Chain treatment, the Poisson variate for the coarse timestep is defined as the sum of two Poisson variates for fine timesteps.

This sub-division of coarse path random variable into the sum of two fine path random variables should work in many settings. The harder step in more general applications is likely to be the second step in the Markov Chain treatment, tightly coupling the increments used for the fine and coarse paths over the same fine timestep.

The general statement of this problem is the following: given two very similar scalar probability distributions, we want to obtain samples  $Z_f, Z_c$  from each in a way which minimises  $\mathbb{E}[|Z_f - Z_c|^p]$ . This corresponds precisely to the Wasserstein metric which defines the “distance” between two probability distributions as

$$\left( \inf_{\gamma} \int \|Z_f - Z_c\|^p d\gamma(Z_f, Z_c) \right)^{1/p},$$

where the minimum is over all joint distributions with the correct marginals. In 1D, the Wasserstein metric is equal to

$$\left( \int_0^1 \left| \Phi_f^{-1}(u) - \Phi_c^{-1}(u) \right|^p du \right)^{1/p},$$

where  $\Phi_f$  and  $\Phi_c$  are the cumulative probability distributions for  $Z_f$  and  $Z_c$  [8], and this minimum is achieved by choosing  $Z_f = \Phi_f^{-1}(U)$ ,  $Z_c = \Phi_c^{-1}(U)$ , for the same uniform  $[0, 1]$  random variable  $U$ . This suggests this may be a good general technique for future multilevel applications, provided one is able to invert the relevant cumulative distributions, possibly through generating appropriate spline approximations.

## 7 Other uses of multilevel

### 7.1 Nested simulation

The pricing of American options is one of the big challenges for Monte Carlo methods in computational finance, and Belomestny & Schoenmakers have recently written a very interesting paper on the use of multilevel Monte Carlo for this purpose [7]. Their method is based on Anderson & Broadie's dual simulation method [1] in which a key component at each timestep in the simulation is to estimate a conditional expectation using a number of sub-paths.

In their multilevel treatment, Belomestny & Schoenmakers use the same uniform timestep on all levels of the simulation. The quantity which changes between different levels of simulation is the number of sub-samples used to estimate the conditional expectation. To couple the coarse and fine levels, the fine level uses  $N_\ell$  sub-samples, and the coarse level uses  $N_{\ell-1} = N_\ell/2$  of them.

Related unpublished research by N. Chen for a similar multilevel treatment of nested simulation found that the multilevel correction variance is reduced if the payoff on the coarse level is replaced by an average of the payoffs obtained using the first  $N_\ell/2$  and the second  $N_\ell/2$  samples. This is similar in some ways to the antithetic approach described earlier.

In future research, Belomestny & Schoenmakers intend to also change the number of timesteps on each level, to increase the overall computational benefits of the multilevel approach.

### 7.2 Truncated series expansions

Building on earlier work by Broadie & Kaya [11], Glasserman & Kim have recently developed an efficient method [30] of exactly simulating the Heston stochastic volatility model [38]. The key to their algorithm is a method of representing the integrated volatility over a time interval  $[0, T]$ , conditional on the initial and final values,  $v_0$  and  $v_T$  as

$$\left( \int_0^T V_s ds \mid V_0 = v_0, V_T = v_T \right) \stackrel{d}{=} \sum_{n=1}^{\infty} x_n + \sum_{n=1}^{\infty} y_n + \sum_{n=1}^{\infty} z_n$$

where  $x_n, y_n, z_n$  are independent random variables.

In practice, they truncate the series expansions at a level which ensures the desired accuracy, but a more severe truncation would lead to a tradeoff between accuracy and computational cost. This makes the algorithm a candidate for a multilevel treatment in which the level  $\ell$  computation performs the truncation at  $N_\ell$ , so the level  $\ell$  computation would use

$$\sum_{n=1}^{N_\ell} x_n + \sum_{n=1}^{N_\ell} y_n + \sum_{n=1}^{N_\ell} z_n$$

while the level  $\ell-1$  computation would truncate the summations at  $N_{\ell-1}$ , but would use the same random variables  $x_n, y_n, z_n$  for  $1 \leq n \leq N_{\ell-1}$ .

This kind of multilevel treatment has not been tested experimentally, but it seems that it might yield some computational savings even though Glasserman & Kim typically only need to retain 10 terms in their summations through the use of a carefully constructed estimator for the truncated remainder. The savings may be larger in other circumstances which require more terms to be retained for the desired accuracy.

### 7.3 *Mixed precision arithmetic*

The final example of the use of multilevel is unusual, because it concerns the computer implementation of Monte Carlo algorithms. In the latest CPUs from Intel and AMD, each core has a vector unit which can perform 8 single precision or 4 double precision operations with one instruction. Also, double precision data takes twice as much time to transfer as single precision data. Hence, single precision computations can be twice as fast as double precision on CPUs, and the difference can be even greater on GPUs. This raises the question of whether single precision arithmetic is sufficient for Monte Carlo simulation.

My view is that it usually is since the finite precision rounding errors are smaller than the other sources of error: statistical error due to Monte Carlo sampling; bias due to SDE discretisation; model uncertainty. However, there can be significant errors when averaging unless one uses binary tree summation [39] to perform the summation, and in addition computing sensitivities by perturbing input parameters (so-called “bumping”) can greatly amplify the rounding errors.

The best solution is perhaps to use double precision for the final averaging, and pathwise sensitivity analysis or the likelihood ratio method for computing sensitivities, but if there remains a need for the path simulation to be performed in double precision then one could use the two-level MLMC approach in which level 0 corresponds to single precision and level 1 corresponds to double precision, with the same random numbers being used for both.

### 7.4 *Multiple outputs*

In all of the discussion so far, we have been concerned with a single expectation arising from a stochastic simulation. However, there are often times when one wishes to estimate the expected value of multiple outputs.



Extending the analysis in section 1.2, when using multilevel to estimate  $M$  different expectations, using  $N_\ell$  samples on each level, the goal is to achieve an acceptably small variance for each output

$$\sum_{\ell=0}^L N_\ell^{-1} V_{\ell,m} \leq \varepsilon_m^2, \quad m = 1, \dots, M,$$

with the desired accuracy  $\varepsilon_m$  being allowed to vary from one output to another, and to do so with the minimum computational cost which is given as usual as

$$\sum_{\ell=0}^L N_\ell C_\ell,$$

assuming that the cost of computing the output functions is negligible compared to the cost of obtaining the stochastic sample (e.g. through an SDE path simulation).

This leads naturally to a constrained optimisation problem with a separate Lagrange multiplier for each output. However, a much simpler idea, due to Tigran Nagapetyan, which in practice is almost always equivalent, is to define

$$V_\ell = \max_m \frac{V_{\ell,m}}{\varepsilon_m^2}$$

and make the variance constraint  $\sum_{\ell=0}^L N_\ell^{-1} V_\ell \leq 1$ .

This is sufficient to ensure that all of the individual constraints are satisfied, and we can then use the standard approach with a single Lagrange multiplier. This multi-output approach is currently being investigated by Nagapetyan, Ritter and the author for the approximation of cumulative distribution functions and probability density functions arising from stochastic simulations.

## 8 Conclusions

In the past 6 years, considerable progress has been achieved with the multilevel Monte Carlo method for a wide range of applications. This review has attempted to emphasise the conceptual simplicity of the multilevel approach; in essence it is simply a recursive control variate strategy, using cheap approximations to some random output quantity as a control variate for more accurate but more costly approximations.

In practice, the challenge is to develop a tight coupling between successive approximation levels, to minimise the variance of the difference in the output obtained from each level. In the context of SDE and SPDE simulations, strong convergence properties are often relied on to obtain a small variance between coarse and fine simulations. In the specific context of a digital option associated with a Brownian SDE,

three treatments were described to effectively smooth the output: a analytic conditional expectation, a “splitting” approximation, and a change of measure. Similar treatments have been found to be helpful in other contexts.

Overall, multilevel methods are being used for an increasingly wide range of applications. The biggest savings are in situations in which the coarsest approximation is very much cheaper than the finest. So far, this includes multi-dimensional SPDEs, and chemical stochastic simulations with 1000’s of timesteps. In SDE simulations which perhaps only require 32 timesteps for the desired level of accuracy, the potential savings are naturally quite limited.

Although this is primarily a survey article, a few new ideas have been introduced:

- equation (1) giving the total computational cost required for a general unbiased multilevel estimator is new, as is the discussion which follows it, although the underlying analysis is not;
- based on the 1D Wasserstein metric, it seems that inverting the relevant cumulative distributions may be a good way to couple fine and coarse level simulations in multilevel implementations;
- the multilevel approach could be used in applications which involve the truncation of series expansions;
- a two-level method combining single and double precision computations might provide useful savings, due to the lower cost of single precision arithmetic;
- a multilevel approach for situations with multiple expectations to be estimated.

Looking to the future, exciting areas for further research include:

- more use of multilevel for nested simulations;
- further investigation of multilevel quasi-Monte Carlo methods;
- continued research on numerical analysis, especially for SPDEs;
- development of multilevel estimators for new applications.

For further information on multilevel Monte Carlo methods, see the webpage [http://people.maths.ox.ac.uk/gilesm/mlmc\\_community.html](http://people.maths.ox.ac.uk/gilesm/mlmc_community.html) which lists the research groups working in the area, and their main publications.

## References

1. L. Andersen and M. Broadie. A primal-dual simulation algorithm for pricing multi-dimensional American options. *Management Science*, 50(9):1222–1234, 2004.
2. D. Anderson and D.J. Higham. Multi-level Monte Carlo for continuous time Markov chains with applications in biochemical kinetics. *SIAM Multiscale Modeling and Simulation*, 10(1):146–179, 2012.
3. A. Asmussen and P. Glynn. *Stochastic Simulation*. Springer, New York, 2007.
4. R. Avikainen. On irregular functionals of SDEs and the Euler scheme. *Finance and Stochastics*, 13(3):381–401, 2009.
5. A. Barth and A. Lang. Multilevel Monte Carlo method with applications to stochastic partial differential equations. *Int. Journal of Computer Mathematics*, 89(18):2479–2498, 2012.

6. A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numerische Mathematik*, 119(1):123–161, 2011.
7. D. Belomestny and J. Schoenmakers. Multilevel dual approach for pricing American style derivatives. *Finance and Stochastics*, 2013.
8. P.J. Bickel and D.A. Freedman. Some asymptotic theory for the bootstrap. *Annals of Statistics*, 9:1196–1217, 1981.
9. A. Brandt, M. Galun, and D. Ron. Optimal multigrid algorithms for calculating thermodynamic limits. *Journal of Statistical Physics*, 74(1-2):313–348, 1994.
10. A. Brandt and V. Ilyin. Multilevel Monte Carlo methods for studying large scale phenomena in fluids. *Journal of Molecular Liquids*, 105(2-3):245–248, 2003.
11. M. Broadie and O. Kaya. Exact simulation of stochastic volatility and other affine jump diffusion processes. *Operations Research*, 54(2):217–231, 2006.
12. S. Burgos and M.B. Giles. Computing Greeks using multilevel path simulation. In L. Plaskota and H. Woźniakowski, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*, pages 281–296. Springer-Verlag, 2012.
13. J. Charrier, R. Scheichl, and A. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM Journal on Numerical Analysis*, 51(1):322–352, 2013.
14. J.M.C. Clark and R.J. Cameron. The maximum rate of convergence of discrete approximations for stochastic differential equations. In B. Grigelionis, editor, *Stochastic Differential Equations*, volume 25 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, 1980.
15. K.A. Cliffe, M.B. Giles, R. Scheichl, and A. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.
16. R. Cont and P. Tankov. *Financial modelling with jump processes*. CRC Press, 2004.
17. S. Dereich. Multilevel Monte Carlo algorithms for Lévy-driven SDEs with Gaussian correction. *Annals of Applied Probability*, 21(1):283–311, 2011.
18. S. Dereich and F. Heidenreich. A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations. *Stochastic Processes and their Applications*, 121(7):1565–1587, 2011.
19. C.R. Dietrich and G.H. Newsam. Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing*, 18(4):10881107, 1997.
20. M.B. Giles. Improved multilevel Monte Carlo convergence using the Milstein scheme. In A. Keller, S. Heinrich, and H. Niederreiter, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 343–358. Springer-Verlag, 2008.
21. M.B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.
22. M.B. Giles. Multilevel Monte Carlo for basket options. In M.D. Rossetti, R.R. Hill, B. Johansson, A. Dunkin, and R.G. Ingalls, editors, *Proceedings of the 2009 Winter Simulation Conference*, pages 1283–1290. IEEE, 2009.
23. M.B. Giles, K. Debrabant, and A. Röbler. Numerical analysis of multilevel Monte Carlo path simulation using the Milstein discretisation. *ArXiv preprint: 1302.4676*, 2013.
24. M.B. Giles, D.J. Higham, and X. Mao. Analysing multilevel Monte Carlo for options with non-globally Lipschitz payoff. *Finance and Stochastics*, 13(3):403–413, 2009.
25. M.B. Giles and C. Reisinger. Stochastic finite differences and multilevel Monte Carlo for a class of SPDEs in finance. *SIAM Journal of Financial Mathematics*, 3(1):572–592, 2012.
26. M.B. Giles and L. Szpruch. Antithetic multilevel Monte Carlo estimation for multidimensional SDEs without Lévy area simulation. *ArXiv preprint: 1202.6283*, 2012.
27. M.B. Giles and L. Szpruch. Antithetic multilevel Monte Carlo estimation for multidimensional SDEs. In *Monte Carlo and Quasi-Monte Carlo Methods 2012 (submitted)*. Springer-Verlag, 2013.
28. M.B. Giles and B.J. Waterhouse. Multilevel quasi-Monte Carlo path simulation. In *Advanced Financial Modelling*, Radon Series on Computational and Applied Mathematics, pages 165–181. De Gruyter, 2009.

29. P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, New York, 2004.
30. P. Glasserman and K.-K. Kim. Gamma expansion of the Heston stochastic volatility model. *Finance and Stochastics*, 15(2):267–296, 2011.
31. P. Glasserman and N. Merener. Convergence of a discretization scheme for jump-diffusion processes with state-dependent intensities. *Proc. Royal Soc. London A*, 460:111–127, 2004.
32. S. Graubner. Multi-level Monte Carlo Methoden für stochastische partielle Differentialgleichungen. Diplomarbeit, TU Darmstadt, 2008.
33. S. Heinrich. Monte Carlo complexity of global solution of integral equations. *Journal of Complexity*, 14(2):151–175, 1998.
34. S. Heinrich. The multilevel method of dependent tests. In N. Balakrishnan, V.B. Melas, and S. Ermakov, editors, *Advances in Stochastic Simulation Methods*, pages 47–61. Springer-Verlag, 2000.
35. S. Heinrich. *Multilevel Monte Carlo Methods*, volume 2179 of *Lecture Notes in Computer Science*, pages 58–67. Springer-Verlag, 2001.
36. S. Heinrich. Monte Carlo approximation of weakly singular integral operators. *Journal of Complexity*, 22(2):192–219, 2006.
37. S. Heinrich and E. Sindambiwe. Monte Carlo complexity of parametric integration. *Journal of Complexity*, 15(3):317–341, 1999.
38. S.I. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6(2):327–343, 1993.
39. N.J. Higham. The accuracy of floating point summation. *SIAM Journal on Scientific Computing*, 14(4):783–799, 1993.
40. H. Hoel, E. von Schwerin, A. Szepessy, and R. Tempone. Adaptive multilevel Monte Carlo simulation. In B. Engquist, O. Runborg, and Y.-H.R. Tsai, editors, *Numerical Analysis of Multiscale Computations*, number 82 in *Lecture Notes in Computational Science and Engineering*, pages 217–234. Springer-Verlag, 2012.
41. A. Kebaier. Statistical Romberg extrapolation: a new variance reduction method and applications to options pricing. *Annals of Applied Probability*, 14(4):2681–2705, 2005.
42. P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, Berlin, 1992.
43. H. Marxen. The multilevel Monte Carlo method used on a Lévy driven SDE. *Monte Carlo Methods and Applications*, 16(2):167–190, 2010.
44. R.C. Merton. Option pricing when underlying stock returns are discontinuous. *Journal of Finance*, 3:125–144, 1976.
45. S. Mishra, C. Schwab, and J. Sukys. Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. *Journal of Computational Physics*, 231(8):3365–3388, 2012.
46. T. Müller-Gronbach and K. Ritter. Variable subspace sampling and multi-level algorithms. In P. L’Ecuyer and A. Owen, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2008*, pages 131–156. Springer-Verlag, 2009.
47. E. Platen and N. Bruti-Liberati. *Numerical Solution of Stochastic Differential Equations With Jumps in Finance*. Springer, 2010.
48. W. Schoutens. *Lévy processes in finance: pricing financial derivatives*. Wiley, 2003.
49. A.L. Speight. A multilevel approach to control variates. *Journal of Computational Finance*, 12:1–25, 2009.
50. A.L. Speight. Multigrid techniques in economics. *Operations Research*, 58(4):1057–1078, 2010.
51. A. Teckentrup, R. Scheichl, M.B. Giles, and E. Ullmann. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numerische Mathematik*, 2013.
52. Y. Xia and M.B. Giles. Multilevel path simulation for jump-diffusion SDEs. In L. Plaskota and H. Woźniakowski, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*, pages 695–708. Springer-Verlag, 2012.